(19) JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **2001117769 A**

(43) Date of publication of application: **27.04.01**

(51) Int. Cl **G06F 9/06**

(21) Application number: **11297759**

(22) Date of filing: **20.10.99**

(71) Applicant: **MATSUSHITA ELECTRIC IND CO LTD**

(72) Inventor: **KANAMARU TOMOKAZU**

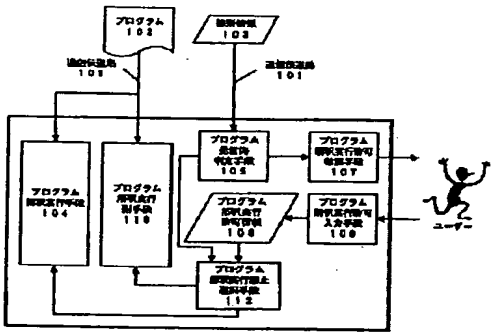
# (54) PROGRAM EXECUTING DEVICE

(57) Abstract:

PROBLEM TO BE SOLVED: To realize the assurance of safety, secrecy and completeness of a downloaded program conveniently at low cost in electrical household appliances having a network function for program distribution.

SOLUTION: This program executing device is constituted of a program interpretation executing means 104, a program interpretation executing sub-means 110, a program transmitting destination judging means 105, an execution permission of program interpretation input means 109 and an execution prohibition of program interpretation selecting means 112. When a judgment result of the program transmitting destination judging means 105 is affirmative, the execution prohibition of program interpretation selecting means 112 permits the program interpretation executing means 104 to interpret and execute a program 102 and when the judgment result is negative, the execution prohibition of program interpretation selecting means 112 prohibits the program interpretation executing means 104 to interpret and execute the program 102.





プログラム実行装置・その4

(19)日本国特許庁（ＪＰ）　(12) 公開特許公報（A）　(11)特許出願公開番号
特開2001－117769
（P2001－117769A）
(43)公開日　平成13年４月27日(2001.4.27)

| (51)Int.Cl.⁷ | 識別記号 | ＦＩ | ﾃｰﾏｺｰﾄﾞ(参考) |
|---|---|---|---|
| Ｇ０６Ｆ　9/06 | ５５０ | Ｇ０６Ｆ　9/06 | ５５０Ｚ　５Ｂ０７６ |

審査請求　未請求　請求項の数８　ＯＬ　（全 17 頁）

(21)出願番号　特願平11－297759

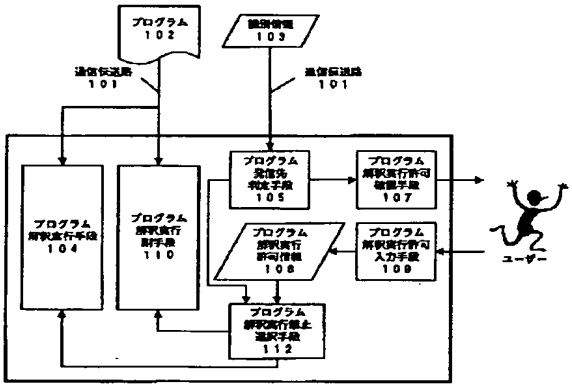
(22)出願日　平成11年10月20日(1999.10.20)

(71)出願人　000005821
松下電器産業株式会社
大阪府門真市大字門真1006番地
(72)発明者　金丸　智一
大阪府門真市大字門真1006番地　松下電器産業株式会社内
(74)代理人　100097445
弁理士　岩橋　文雄　（外２名）

Ｆターム(参考)　5B076 BB06 CA07 FD00

(54)【発明の名称】　プログラム実行装置

(57)【要約】
【課題】　プログラム配信のためのネットワーク機能を備えた家電機器において、ダウンロードしたプログラムの安全性・秘匿性・完全性の保証を、低コストかつ簡便に実現する。
【解決手段】　プログラム解釈実行手段１０４と、プログラム解釈実行副手段１１０と、プログラム発信先判定手段１０５と、プログラム解釈実行許可入力手段１０９と、プログラム解釈実行禁止選択手段１１２とから構成される。プログラム発信先判定手段１０５の判定結果が是である場合には、プログラム解釈実行禁止選択手段１１２は、プログラム解釈実行手段１０４がプログラム１０２を解釈実行することを許可し、判定結果が非である場合には、プログラム解釈実行禁止選択手段１１２は、プログラム解釈実行手段１０４がプログラム１０２を解釈実行することを禁止する。




プログラム実行装置・その4

1

【特許請求の範囲】

【請求項１】　外部伝送路を通して送信される、プログラムと外部伝送路を通して送信される、プログラムの発信元の識別情報とを入力とする、プログラム実行装置であり、

プログラム実行装置は、

プログラムを解釈実行する手段である、プログラム解釈実行手段と、

識別情報が、信頼されたプログラム発信元を示すものであるか否かを判定する手段である、プログラム発信先判定手段と、

プログラム解釈実行禁止手段と、

からなり、

プログラム発信先判定手段の判定に従って、プログラム解釈実行禁止手段は、プログラム解釈実行手段がプログラムを解釈実行するのを禁止することを特徴とする、プログラム実行装置。

【請求項２】　外部伝送路を通して送信される、プログラムと外部伝送路を通して送信される、プログラムの発信元の識別情報とを入力とする、プログラム実行装置であり、

プログラム実行装置は、

前記プログラム解釈実行手段と、

前記プログラム発信先判定手段と、

前記プログラム解釈実行禁止手段と、

プログラムの解釈実行を許可する、あるいは、禁止する、のいずれかを示す、解釈実行許可情報を、ユーザが入力する手段である、プログラム解釈実行許可入力手段と、

からなり、

解釈実行許可情報に従って、プログラム解釈実行禁止手段は、プログラム解釈実行手段がプログラムを解釈実行するのを禁止することを特徴とする、プログラム実行装置。

【請求項３】　外部伝送路を通して送信される、プログラムと外部伝送路を通して送信される、プログラムの発信元の識別情報とを入力とする、プログラム実行装置であり、

プログラム実行装置は、

前記プログラム解釈実行手段と、

プログラムを解釈実行する第２の手段である、プログラム解釈実行副手段と、

前記プログラム発信先判定手段と、

プログラム解釈実行選択手段と、

からなり、

プログラム発信先判定手段の判定に従ってプログラム解釈実行選択手段が、プログラムがプログラム解釈実行手段で実行されるのを許可するかあるいは禁止するかを選択して、プログラムがプログラム解釈実行手段で実行されるのを禁止する場合には、プログラムをプログラム解釈実行手段副手段で実行させることを特徴とする、プログラム実行装置。

2

【請求項４】　外部伝送路を通して送信される、プログラムと外部伝送路を通して送信される、プログラムの発信元の識別情報とを入力とする、プログラム実行装置であり、

プログラム実行装置は、

前記プログラム解釈実行手段と、

前記プログラム解釈実行副手段と、

前記プログラム発信先判定手段と、

プログラムの解釈実行を許可する、あるいは、禁止する、のいずれかを示す、前記解釈実行許可情報を、ユーザが入力する手段である、前記プログラム解釈実行許可入力手段と、

プログラム解釈実行禁止選択手段と、

からなり、

プログラム発信先判定手段の判定結果が是である場合には、プログラム解釈実行禁止選択手段は、プログラム解釈実行手段がプログラムを解釈実行することを許可し、

プログラム発信先判定手段の判定結果が非である場合には、プログラム解釈実行禁止選択手段は、プログラム解釈実行手段がプログラムを解釈実行することを禁止することを特徴とし、

プログラム発信先判定手段の判定結果が非である場合には、プログラム解釈実行禁止選択手段はは前記解釈実行許可情報に従って、プログラム解釈実行副手段がプログラムを解釈実行することを許可する、あるいは禁止する、のいずれかを選択することを特徴とする、プログラム実行装置。

【請求項５】　プログラム発信先判定手段の判定結果が否である場合、プログラムの解釈実行を許可するか否かをユーザに問い合わせる手段である、プログラム解釈実行許可確認手段を備えることを特徴とした、請求項２および請求項４のプログラム実行装置。

【請求項６】　識別情報はＩＰアドレスを含み、プログラム発信先判定手段は、ＩＰアドレスを用いて判定を行うことを特徴とする、請求項１、２、３、４、５記載のプログラム実行装置。

【請求項７】　識別情報はＵＲＬを含み、プログラム発信先判定手段１０５は、ＵＲＬを用いて判定を行うことを特徴とする、請求項１、２、３、４、５記載のプログラム実行装置。

【請求項８】　識別情報は暗号化された署名を含み、プログラム発信先判定手段は、暗号化された署名を用いて判定を行うことを特徴とする、請求項１、２、３、４、５記載のプログラム実行装置。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明はネットワーク通信によりプログラムをダウンロードし、プログラムを解釈実

行する環境、特に家電機器など、使用目的が特化されたプログラムを解釈実行する環境において、ダウンロードされたプログラムの検証を行なう装置およびその方法に関する。

【０００２】

【従来の技術】近年、家電機器に対するプログラム配信のためのネットワーク機能搭載の要求が高まっている。

【０００３】ネットワークを通じてプログラムの配信が行われることで、例えば、家電機器に対する機能の更新や追加が容易になり、新たなサービスの提供が可能になる、などの多くの利点が生まれる。

【０００４】既にコンピューティングの分野では、こういったプログラムの配信は頻繁に行なわれている。

【０００５】その代表的なものに、インターネットのウェブサーバと、ブラウザを載せたＰＣの間のネットワークがある。このネットワークにおいては、ネットワークに接続する者であれば誰でも、そこに流通するコンテンツを自由にダウンロードして利用できる。ここではこの性質を持つネットワークを「開かれたネットワーク」と呼ぶことにする。

【０００６】開かれたネットワークの元では、利用者はネットワークに存在するプログラムもまた、自由にダウンロードし、実行することができる。

【０００７】しかし一方で、開かれたネットワークでは、以下のことをネットワーク利用者によるリスクの分担の上で行なうという特徴を持つ。１．接続先が意図した相手であること（認証と許可）。２．コンテンツが伝送経路で改ざんや盗聴がなされないこと（完全性と秘匿性）。３．コンテンツがそれを利用する環境で危険な振る舞いをしないこと（安全性）。

【０００８】つまり開かれたネットワークとは、上記の１．２．３．が完全に保証される環境ではない。そのため、悪意ある第３者によって作成された危険なプログラムを、そうとは知らずに利用者がダウンロードし、実行してしまう可能性がある。

【０００９】一方、開かれたネットワークとは異なり、流通するコンテンツをダウンロードする際に、コンテンツの暗号化やコンテンツをダウンロードする際の利用者の認証情報の確認などの必要を課すことによって、上記の１．２．３．の点を保証するネットワークも存在する。この性質を持つネットワークを「閉じられたネットワーク」と呼ぶことにする。

【００１０】家電機器には一般的に高い信頼性・安全性が求められており、家電機器に対するプログラム配信の際には、信頼できる配信元が作成した安全なプログラムのみがダウンロードされることが求められる。

【００１１】加えて家電機器に対するプログラム配信には、しばしば課金やデータの保護などが安全・正確に行われることが必要である。これを実現するには開かれたネットワークでは不充分である。これらの観点から、家



電機器に対するコンテンツの配信は、閉じられたネットワークが使用されることが望ましい。

【００１２】実際、現行の家電機器向けを想定したネットワークは、何らかの手段によって上記の１．２．３．の点を保証した、閉じられたネットワークであるものが殆どである。

【００１３】例えば現行の電話を用いた各種のサービスも、電話のネットワークを利用し、通話によるネットワークの利用に伴う課金を正確に行なう必然性から、そのサービスは閉じられたネットワークで実施されている。もちろん、コンテンツに暗号化等の処理を加えることにより、より強固で安全なネットワークとすることも可能である。

【００１４】

【発明が解決しようとする課題】しかしながら近年、家電機器分野においても、インターネットに接続可能な携帯電話などのサービスに代表されるように、開かれたネットワーク環境に接続することが、一部の機器では可能になりつつある。インターネットに代表される開かれたネットワークと、家電機器との親和性は、今後さらに高まっていくであろうことが予想される。

【００１５】このような状況においては先に述べたように、閉じられたネットワークを用い、信頼できる発信元からのみ、保証されたプログラムをダウンロードし実行することを、家電機器に遵守させることは難しくなる。

【００１６】一つの解決策は、プログラムを解釈実行する実行環境のほうに、プログラムが危険なものでないかどうかを検査する検証機構を持たせる、というものである。

【００１７】この解決策をとっている代表的なアーキテクチャにはＪａｖａがある。

【００１８】開発言語としてのＪａｖａ言語は、オブジェクト指向言語としての開発効率の高さ、バグが出難い言語仕様、移植性の容易さ、既存の開発環境の豊富さ、など多くの優れた点を持っている。

【００１９】さらにＪａｖａアーキテクチャは、Ｊａｖａ仮想マシンというインタプリタ機構によりプログラムを解釈実行する、という仕組みを持ち、機器のハードウェアやＯＳに依存しないプログラムを開発できるため、家電分野においてプログラム配信を実現するのに有望なアーキテクチャとみなされている。

【００２０】Ｊａｖａは、ネットワークを用いたプログラム（コンテンツ）配信の際に、セキュリティに関して高い安全性を保持していることでも知られている。具体的にはＪａｖａは、以下のような検証機構を持っている。

【００２１】（ＬａｕｒａＬｅｍａｙ，ｃｈａｒｌｅｓＬ．Ｐｅｒｋｉｎｓ，"ＴｅａｃｈＹｏｕｒｓｅｌｆＪａｖａｉｎ２１ｄａｙｓ"ＰｒｅｎｔｉｃｅＨａｌｌ，１９９６）１．Ｊａｖａ言語そのものが安全性の高い言

語仕様を持っており、Ｊａｖａ言語によって悪意ある変造コードを作成することは困難になっている。例えば、メモリに対する参照値などを、プログラマは直接操作することはできない。２．Ｊａｖａ仮想マシンは、実行形式（Ｊａｖａクラスファイル）が正しい規則に沿って記録されているかどうかを検証する機構を備える。これにより、変造された実行形式を実行解釈することを、Ｊａｖａ仮想マシンは拒絶する。３．仮想マシンのプログラム・ローダは記憶範囲をグループ別に管理しており、「保護の弱い」記憶範囲に記録されたクラスファイル（実行形式）が、「より保護の強い」記憶範囲に記録されたクラスファイルを侵害したり、許可なく参照したりすることはできない。例えば、インターネットから配信されたクラスファイルが、ローカルな領域に記録されていたクラスファイルに、許可なく置き換えられたりすることはない。４．ＪａｖａＡＰＩが提供する機能群は、プログラムが破壊的な動作を行なえないように設計されている。

【００２２】以上のようにＪａｖａは、開発言語からプログラム実行系までの各レベルにおいて、強固な検証機構を有しているが、このような機能を持たせたため、Ｊａｖａ仮想マシンはその動作のために多大な使用メモリ量と高性能のプロセッサを必要としている。

【００２３】例えば２を実現するために、仮想マシンは実行形式中のＪａｖａの命令コードに対してフロー解析（Ａ．Ａｈｏ，Ｒ．Ｓｅｔｈｉ，Ｊ．Ｕｌｌｍａｎ，"ｃｏｍｐｉｌｅｒＰｒｉｎｃｉｐｌｅｓ，Ｔｅｃｈｉｎｉｑｕｅｓ，ａｎｄＴｏｏｌｓ"，Ａｄｄｉｓｏｎ－Ｗｓｅｌｅｙ，１９８６）を行わなければならず、これはＪａｖａ仮想マシンの実装コードサイズと使用メモリ量の増大を招く（試算によると、２．の検証機構を備えない場合の総使用メモリ量と比較して、１．５～３．０倍増大する）。

【００２４】さらにこれらの検証処理は、プログラムの実行中にもしばしば発生するものであるため、Ｊａｖａ仮想マシンの処理の負荷を大きくし、高速にプログラムを動作させることへの妨げにもなる。

【００２５】家電機器分野ではコストに対する要求は非常に厳しく、コンピューティング分野以上に省資源・低コスト・低消費電力が強く要求されるものが大多数である。

【００２６】例えば家電機器にＪａｖａを搭載させるアプローチの代表的なものに、サンマイクロシステムズが家電組み込み向けに作成した仕様であるＥｍｂｅｄｄｅｄＪａｖａアーキテクチャがあるが、プログラムの動作のためには最低、ＲＡＭ５１２Ｋバイト、ＲＯＭ５１２Ｋバイト、２５ＭＨｚ以上のＣＰＵを必要としている（ＤｅｅｐａｋＭｕｌｃｈａｎｄａｎｉ，"ＪａｖａｆｏｒＥｍｂｅｄｄｅｄＳｙｓｔｅｍｓ"，ＩＥＥＥＩＮＴＥＲＮＥＴＣＯＭＰＵＴＩＮＧ，Ｍａｙ－ｊｕｎｅ１９９８）。これはハイエンドの情報家電機器に対して搭載可能、というレベルを実現したに留まるものであり、必要メモリ量は依然として一般の家電機器にとっては大きなサイズである。



【００２７】特に、携帯電話や白物家電などには、低動作周波数の８／１６ｂｉｔマイコンや、数十ＫＢ程度のメモリしか搭載しないものが主流であり、このような機器に対してはＥｍｂｅｄｄｅｄＪａｖａを適用することは不可能である。

【００２８】以上に示される通り、Ｊａｖａに代表されるような、検証機構を実行環境に搭載してネットワークを流れるプログラムの安全性を保証するというアプローチは、一般の家電機器全般に適用するには非常に困難である。

【００２９】本発明は、プログラム配信のためのネットワーク機能を備えた家電機器における問題である、ダウンロードしたプログラムの安全性・秘匿性・完全性の保証を、可能な限り低コストで簡便に実現するための手段を提供するものである。

【００３０】

【課題を解決するための手段】本発明の装置は、外部伝送路を通して、プログラムと、プログラムの発信元の識別情報を受信する。装置は受信した識別情報を判定し、その結果により、プログラムが安全なものであるか、それとも安全性の保障できない、実行環境とユーザに危険を及ぼす可能性のあるものであるか、を判断する。

【００３１】安全と判断したプログラムに対しては、解釈実行を許可する。この解釈実行の手段はプログラムに対して特別に検証機構による処理を必要としないものを想定している。

【００３２】よってプログラムを、使用メモリ量を少なく抑えたまま、高速に動作させることができる。

【００３３】安全性が保証できないプログラムに対しては、本発明は以下のような対処方法を提供している。

【００３４】１）安全性が保証できないプログラムは一切解釈実行させない。

【００３５】この方法を取る場合、プログラム実行装置は危険なプログラムを一切動作させないため、検証機構を備える必要はない。プログラムの解釈実行のために必要な資源量は小さく抑えたまま、安全と判断されたプログラムは全て高速に動作させることができる。

【００３６】２）ユーザによって与えられた、プログラムの解釈実行を許可する、あるいは禁止する、のいずれかを示す情報を元に、プログラムの解釈実行を決定する。

【００３７】この方法を取る場合、プログラム実行装置は安全性が保証できないプログラムを動作させるかどうかをユーザの判断に任せる。

【００３８】本来家電機器はその特徴上、信頼できる特定の（おそらくは機器を製造したメーカーの保証を受け

た）発信元からのみ、プログラムをダウンロードし実行することを推奨するものである。そのような背景においても、ユーザが開かれたネットワークから、プログラム（コンテンツ）のダウンロードを行うというのは、例外的な状況下においてであると考えられる。

【００３９】つまり、ユーザにとって何らかの理由によって有益と思われるプログラム・コンテンツを、その機器を使用して獲得したい場合である。

【００４０】開かれたネットワークから、プログラム（コンテンツ）を獲得する場合、ユーザはそのダウンロード対象プログラム・コンテンツが安全なものであるとの知識を持っていることが前提であり、ユーザは自己の責任においてその行為を行わなければならない。

【００４１】２）の方法は、そのユーザの判断を、プログラム実行装置に反映させる方式である。

【００４２】プログラムはユーザの自己責任によってプログラムを解釈実行させるため、検証処理は行われないことが前提となり、プログラムの解釈実行のために必要な資源量は小さく抑えたまま、プログラムを高速に動作させることができる。

【００４３】３）ダウンロードしたプログラムが危険なものである可能性があることをユーザに知らせた後、２）を行う。

【００４４】２）と同じであるがプログラムを動作させる前に、判断を促すための通告をユーザに対して行うことができる。

【００４５】ユーザはそれを用い、より正確な判断で、プログラムの解釈実行の許可や禁止の情報を装置に対して与えることができる。

【００４６】４）プログラム実行装置に、プログラムの解釈実行のための、従来とは別の第２の手段を用意し、安全性が保証できないプログラムはその第２の手段により実行することにする。

【００４７】安全性が保障されるプログラムとされないプログラムの解釈実行のための手段を分離する方法である。

【００４８】この場合、安全性が保証されないプログラムを解釈実行する手段は、検証処理を動作させつつプログラムを解釈実行すること、あるいは、プログラム実行装置の他の手段には影響を与えず独立して存在していることが望ましい。

【００４９】これにより、安全なプログラムを高速に動作させることを保証したまま、危険なプログラムもまた動作させることができる。５）２）と４）の方式を同時にとるもの６）３）と４）の方式を同時にとるもの本発明はこれらの手段を、プログラムを実行する機器に対して提供するものである。

【００５０】

【発明の実施の形態】本発明は、外部伝送路に接続されたプログラム実行環境全般に適用可能なものであるが、特に省資源・低コストが要求される家電機器への適用を強く意図したものである。

【００５１】ここでは特に、ディスプレイを装備したインターネットに接続可能な携帯端末に対して、本発明を適用した例を適宜加えながら、説明を行っていく。

【００５２】以下、本発明の実施の形態について、図面を用いて説明する。

【００５３】まず図１は、請求項１に記載のプログラム実行装置の構成を示したものである。

【００５４】図１の装置の処理のフローチャートを図５に示す。

【００５５】これらについて順に説明していく。

【００５６】外部伝送路１０１から受信する、プログラム１０２と識別情報１０３は、例えばインターネットなどのネットワークを通じてダウンロードする。

【００５７】プログラムは、装置が備える実行系により解釈実行が可能な形式を持ったデータとして獲得することができる。例えば装置がＪａｖａ仮想マシンを備えているならば、その装置はＪａｖａの実行形式であるＪａｖａクラスファイルが実行可能であるため、Ｊａｖａクラスファイルを外部伝送路を通じてダウンロードする。

【００５８】識別情報とは、プログラムをダウンロードする時に、例えば、その発信元が特定されるものであるとする。本実施例では、説明を簡単にするために、これをプログラムの名前で代用する。もちろん他の情報であってもよい。

【００５９】識別情報として用いることが可能なものに、例えば、インターネットに接続した機器に固有に割り振られるＩＰアドレスや、Ｗｅｂ，ＦＴＰ，Ｇｏｐｈｅｒ，Ｎｅｗｓ，ＴＥＬＮＥＴなどの情報を指定する世界共通で一意なアドレスであるＵＲＬ（ＵｎｉｆｏｒｍＲｅｓｏｕｒｃｅＬｏｃａｔｏｒ）がある。

【００６０】米国ＮｅｔｓｃａｐｅＣｏｍｍｕｎｉｃａｔｉｏｎｓＣｏｒｐｏｒａｔｉｏｎによるＳＳＬ（ＳｅｃｕｒｅＳｏｃｋｅｔｓＬａｙｅｒ）プロトコルでは、相手の認証や使用する暗号やデジタル署名のアルゴリズムなどに関するネゴシエーションを行ない，相互に認証してから，データの送受信を行っているが、もちろんＳＳＬに使用されるものに代表される暗号化された署名を識別情報とすることも可能である。もちろん、識別情報は上記に挙げたもの以外のものであってもよい。

【００６１】図５において最初の処理であるＰ１０５では、識別情報１０３を解読し、識別情報が、信頼されたプログラム発信元を示しているか否かを判定する。

【００６２】これは図１における、プログラム発信先判定手段１０５によって行われる処理である。

【００６３】この処理Ｐ１０５の具現は、何らかの形で識別情報を解釈して、その信頼性を判定できる必要がある。例えば以下のような形で、これを実現することができる。

【００６４】プログラム発信先判定手段１０５は、信頼できる発信元を示した全ての識別情報を記録したデータを保持している。信頼できるか否かの判定は、ダウンロードしてきた識別情報をこのデータと照合し、識別情報がデータに登録されているなら、この識別情報は信頼できると判定し、登録されていないなら信頼できないと判定する。

【００６５】識別情報として、ＳＳＬプロトコルで用いられるような暗号化された署名を用いている場合には、プログラム発信先判定手段１０５は、暗号の解読に成功するか否かによって、識別情報を信頼できるかできないかを決定することもできる。もちろん、信頼性の判定は他の手段で行うこともできる。

【００６６】処理Ｐ１０４は、プログラム１０２を解釈実行する。

【００６７】これは図１における、プログラム解釈実行手段１０４によって行われる処理である。

【００６８】本発明の備えるプログラム解釈実行手段１０４は、プログラムを解釈実行することが可能な環境が備える、従来の実行系と同じ仕組みであり、従来と同様にプログラムの実行解釈を行う。

【００６９】処理Ｐ１０６は、プログラム１０２がプログラム解釈実行手段１０４によって解釈実行されるのを禁止する。

【００７０】これは図１における、プログラム解釈実行禁止手段１０６によって行われる処理である。

【００７１】プログラム解釈実行禁止手段１０６の実現には、例えば以下のようなものが考えられる。

【００７２】プログラム解釈実行禁止手段１０６は内部に、図９に示されるテーブルを保持する。

【００７３】このテーブルのエントリは、ダウンロードしたプログラムの識別子と、それに対応する、プログラムの状態を示す値とで構成される。プログラムの状態を示す値は、ｐｅｒｍｉｔｅｄもしくはｆｏｒｂｉｄｅｎという２値のどちらかを取る。テーブルに記録される時の初期状態はｐｅｒｍｉｔｔｅｄであるとする。

【００７４】プログラム解釈実行手段１０４は、解釈実行を始める前に図９のテーブルを参照して、解釈実行しようとするプログラムの識別子を検索する。該当するプログラムの識別子のエントリの、プログラムの状態を示す値がｆｏｒｂｉｄｄｅｎであるか、あるいはエントリがテーブルにない場合には、そのプログラムの解釈実行を開始しないことにする。

【００７５】この場合、処理Ｐ１０６は、該当するプログラムの識別子のエントリの、プログラムの状態を示す値をｆｏｒｂｉｄｄｅｎに変更することによって実現できる。

【００７６】ここではプログラム解釈実行禁止手段１０６の実現にテーブルを用いて実施例を示したが、もちろんテーブル以外の構造、例えばリスト、木、ハッシュを用いて実現することも可能である。

【００７７】処理Ｐ１０４、あるいは処理Ｐ１０６が終了すると、図１で行われる処理全体が完了する。

【００７８】図２は、請求項５記載のプログラム実行装置の構成を示したものである。

【００７９】図２から、プログラム解釈実行許可確認手段１０７を除いた図面が、請求項２記載のプログラム実行装置の構成を示すものになる。

【００８０】図２の装置の処理のフローチャートを図６に示す。

【００８１】図６において、Ｐ２０７の処理を「何もしない」に置き換えたものが、請求項２記載の装置の処理のフローチャートになる。

【００８２】これらについて順に説明していく。

【００８３】外部伝送路１０１から受信する、プログラム１０２および識別情報１０３は、先に説明したものと同じである。

【００８４】図６において最初の処理であるＰ２０５は、図２におけるプログラム発信先判定手段１０５によって行われる処理であり、先に説明したＰ１０５と同じ処理を行う。

【００８５】処理Ｐ２０７は、プログラムの実行を許可するか否かをユーザに問い合わせる。

【００８６】これは図２における、プログラム解釈実行許可確認手段１０７によって行われる処理である。

【００８７】例えばこれは、装置が備えるディスプレイに、図１０のような表示を行うことで実現できる。

【００８８】処理Ｐ２０９は、プログラムの実行を許可するか否かを示す情報である、プログラム解釈実行許可情報１０８を得る。

【００８９】これは図２における、プログラム解釈実行許可入力手段１０９によって行われる処理である。

【００９０】装置のユーザは、処理Ｐ２０９が行われる前に、何らかの意思表明を装置に対して行う。プログラム解釈実行許可入力手段は、その意思表明を受け、装置内部のデータとして、プログラム解釈実行許可情報１０８を生成する。

【００９１】ユーザによる意思表明は、装置が備える入力装置、例えばボタン、スイッチ、ダイヤル、マウス、トラックボール、ジョイスティック、等による入力操作の組み合わせによって、あるいはこれらの操作を何もしないことによってなされる。

【００９２】例えば・ユーザは、”機能”ボタンを押した後、”０”ボタンを押す、という行為を、装置に対して行う。これによって装置は、全てのプログラムの実行が許可されたという情報を得る。・ユーザは、ダイヤルによって”ｍａｚｅ．ｃｊ”という文字列をディスプレイ上で選択した後に、”１”ボタンを押す。これによって装置は、”ｍａｚｅ．ｃｊ”という識別名を持つプログラムの実行が禁止されたという情報を得る。・ユーザ

は、処理Ｐ２０９が行われる前に、プログラムの実行の許可／禁止を指示する特定の入力を、何ら装置に対して与えなかったとする。これによって装置は、”ｍａｚｅ．ｃｊ”という識別名を持つプログラムの実行が禁止されたという情報を得る。

【００９３】ユーザの意思表明は、処理Ｐ２０７と連携して実施されても良い。例えば、・図１０の表示が行なわれた後に、”０”ボタンを押す、という行為を、装置に対して行う。これによって装置は、”ｍａｚｅ．ｃｊ”という識別名を持つプログラムの実行が許可されたという情報を得る。

【００９４】ここでは”０”，”１”，”機能”というボタンとダイヤル操作の組み合わせによる実施例を示したが、もちろんボタンの種類や押す順番は上に挙げたもの以外であっても良いし、入力装置の組み合わせも他のものであっても良い。

【００９５】処理Ｐ２０６Ａは、プログラム解釈実行許可情報１０８を解釈して、ユーザがプログラム１０２の実行を許可したか否かを判定する。

【００９６】これは図２において、プログラム解釈実行禁止手段１０６によって行われる処理である。

【００９７】処理Ｐ２０４は、プログラム１０２を解釈実行する。

【００９８】これは図２における、プログラム解釈実行手段１０４によって行われる処理であり、先に説明した処理Ｐ１０４と同じものである。

【００９９】処理Ｐ２０６Ｂは、プログラム１０２の解釈実行を禁止する。

【０１００】この処理は、図２のプログラム解釈実行禁止手段１０６によって行われる処理であり、先に説明した処理Ｐ１０６と同じものである。

【０１０１】処理Ｐ２０４、あるいは処理Ｐ２０６Ｂが終了すると、図２で行われる処理全体が完了する。

【０１０２】図３は、請求項３に記載のプログラム実行装置の構成を示したものである。

【０１０３】図３の装置の処理のフローチャートを図７に示す。

【０１０４】これらについて順に説明していく。

【０１０５】外部伝送路１０１から受信する、プログラム１０２および識別情報１０３は、先に説明したものと同じである。

【０１０６】図７において最初の処理であるＰ３０５は、図３におけるプログラム発信先判定手段１０５によって行われる処理であり、先に説明したＰ１０５と同じ処理を行う。

【０１０７】処理Ｐ３０４は、プログラム１０２をプログラム解釈実行手段１０４によって実行する。

【０１０８】これは図３における、プログラム解釈実行選択手段１１１と、プログラム解釈実行手段１０４によって行われる処理である。



【０１０９】処理Ｐ３１０は、プログラム１０２をプログラム解釈実行副手段１１０によって実行する。

【０１１０】これは図３における、プログラム解釈実行選択手段１１１と、プログラム解釈実行副手段１１０によって行われる処理である。

【０１１１】プログラム解釈実行副手段１１０は、プログラムを解釈実行することが可能な環境が備える、従来の実行系と同じ仕組みであり、従来と同様にプログラムの実行解釈を行うが、これはプログラム解釈実行手段１０４とは別の手段として装置内部に実現される。

【０１１２】プログラム解釈実行選択手段１１１の実現には、例えば以下のようなものが考えられる。

【０１１３】プログラム解釈実行選択手段１１１は内部に、図１１に示されるテーブルを保持する。

【０１１４】このテーブルのエントリは、ダウンロードしたプログラムの識別子と、それに対応する、プログラムの状態を示す値とで構成される。

【０１１５】プログラムの状態を示す値は、ｓｕｃｕｒｅｄもしくはｕｎｓｅｃｕｒｅｄという２値のどちらかを取る。

【０１１６】プログラムの実行の前に、プログラム解釈実行選択手段１１１は、プログラム発信先判定手段１０５の判定結果によって、信頼された発信元からのプログラムであればｓｅｃｕｒｅｄを、そうでなければｕｎｓｅｃｕｒｅｄを記録する。

【０１１７】プログラム解釈実行手段１０４は、解釈実行を始める前に図１１のテーブルを参照して、解釈実行しようとするプログラムの識別子を検索する。該当するプログラムの識別子のエントリの、プログラムの状態を示す値がｓｅｃｕｒｅｄである場合にのみ、そのプログラムの解釈実行を開始する。

【０１１８】プログラムの状態を示す値がｕｎｓｅｃｕｒｅｄであるか、あるいはエントリがテーブルにない場合には、そのプログラムの解釈実行はプログラム解釈実行副手段１１０によって実行される。

【０１１９】ここではプログラム解釈実行選択手段１１１の実現にテーブルを用いて実施例を示したが、もちろんテーブル以外の構造、例えばリスト、木、ハッシュを用いて実現することも可能である。

【０１２０】処理Ｐ３０４、あるいは処理Ｐ３１０が終了すると、図３で行われる処理全体が完了する。

【０１２１】図４は、請求項５記載のプログラム実行装置の構成を示したものである。

【０１２２】図４から、プログラム解釈実行許可確認手段１０７を除いた図面が、請求項４記載のプログラム実行装置の構成を示すものになる。

【０１２３】図４の装置の処理のフローチャートを図８に示す。

【０１２４】図８において、Ｐ４０７の処理を「何もしない」に置き換えたものが、請求項４記載の装置の処理

のフローチャートになる。

【０１２５】これらについて順に説明していく。

【０１２６】外部伝送路１０１から受信する、プログラム１０２および識別情報１０３の説明は、先に説明したものと同じである。

【０１２７】図８において最初の処理であるＰ４０５は、図４におけるプログラム発信先判定手段１０５によって行われる処理であり、先に説明したＰ１０５と同じ処理を行う。

【０１２８】処理Ｐ４０７は、プログラムの実行を許可するか否かをユーザに問い合わせる。

【０１２９】これは図４における、プログラム解釈実行許可確認手段１０７によって行われる処理であり、先に説明したＰ２０７と同じ処理を行う。

【０１３０】処理Ｐ４０９は、プログラムの実行を許可するか否かを示す情報である、プログラム解釈実行許可情報１０８を得る。

【０１３１】これは図４における、プログラム解釈実行許可入力手段１０９によって行われる処理であり、先に説明したＰ２０９と同じ処理を行う。

【０１３２】処理Ｐ４０６Ａは、プログラム解釈実行許可情報１０８を解釈して、ユーザがプログラム１０２の実行を許可したか否かを判定する。

【０１３３】これは図４において、プログラム解釈実行禁止手段１０６によって行われる処理である。

【０１３４】処理Ｐ４０４は、プログラム１０２をプログラム解釈実行手段１０４によって実行する。

【０１３５】これは図４における、プログラム解釈実行禁止選択手段１１２と、プログラム解釈実行手段１０４によって行われる処理である。

【０１３６】処理Ｐ４１０は、プログラム１０２をプログラム解釈実行副手段１１０によって実行する。

【０１３７】これは図４における、プログラム解釈実行禁止選択手段１１２と、プログラム解釈実行副手段１１０によって行われる処理である。

【０１３８】処理Ｐ４０６Ｂは、プログラム１０２の解釈実行を禁止する。

【０１３９】この処理は、図４のプログラム解釈実行禁止選択手段１１２によって行われる処理である。

【０１４０】プログラム解釈実行禁止選択手段１１２の実現には、例えば以下のようなものが考えられる。

【０１４１】プログラム解釈実行禁止選択手段１１２は内部に、図１２に示されるテーブルを保持する。このテーブルのエントリは、ダウンロードしたプログラムの識別子と、それに対応する、プログラムの状態を示す値とで構成される。

【０１４２】プログラムの状態を示す値は、ｓｕｃｕｒｅｄもしくはｐｅｒｍｉｔｔｅｄもしくはｆｏｒｂｉｄｄｅｎという３つの値のいずれかを取る。ここではエントリに最初に記録されるときの初期値はｓｅｃｕｒｅｄであるとする。

【０１４３】処理Ｐ４０６Ａにおいて、プログラム解釈実行選択手段１１１は、プログラム解釈実行許可情報１０８を解釈して、ユーザが実行を許可していれば、プログラムの状態を示す値にｐｅｒｍｉｔｔｅｄを、そうでなければｆｏｒｂｉｄｄｅｎを記録する。

【０１４４】プログラム解釈実行手段１０４は、プログラム１０２の解釈実行を始める前に図１１のテーブルを参照して、解釈実行しようとするプログラムの識別子を検索する。該当するプログラムの識別子のエントリの、プログラムの状態を示す値がｓｅｃｕｒｅｄである場合にのみ、そのプログラムの解釈実行を開始する。

【０１４５】プログラムの状態を示す値がｐｅｒｍｉｔｔｅｄである場合には、プログラム１０２はプログラム解釈実行副手段１１０によって解釈実行される。

【０１４６】プログラムの状態を示す値がｆｏｒｂｉｄｄｅｎであるか、あるいはエントリがテーブルにない場合には、プログラム１０２は解釈実行されない。

【０１４７】ここではプログラム解釈実行禁止選択手段１１２の実現にテーブルを用いて実施例を示したが、もちろんテーブル以外の構造、例えばリスト、木、ハッシュを用いて実現することも可能である。

【０１４８】処理Ｐ４０４、あるいは処理Ｐ４１０、あるいは処理Ｐ４０６Ｂが終了すると、図４で行われる処理全体が完了する。

【０１４９】以上が本発明の実施の形態である。

【０１５０】

【発明の効果】以上のように本発明は、外部伝送路を通して、プログラムと、プログラムの発信元の識別情報を受信し、その識別情報が、信頼できる発信元を示しているかどうかを判定する。

【０１５１】判定の結果、信頼できる発信元のものであることが判別できない場合、実行環境は、そのことをユーザ（実行環境の使用者）に知らせ、ユーザによって与えられた情報を元に、プログラムの解釈実行を決定する。もしくは、従来の解釈実行手段とは異なる、第２の解釈実行手段によって、そのプログラムを解釈実行する。

【０１５２】このような機構を設けることにより、プログラムを実行する装置は、安全であると保証されるプログラムは従来通り、使用資源が少なくかつ高速な解釈実行を行うことができる。かつダウンロードしたプログラムの安全性・秘匿性・完全性の保証を、装置の低コスト・省資源性を保ったまま行うことができる。

【図面の簡単な説明】

【図１】本発明のプログラム実行装置の構成例を示す図

【図２】本発明のプログラム実行装置の構成例を示す図

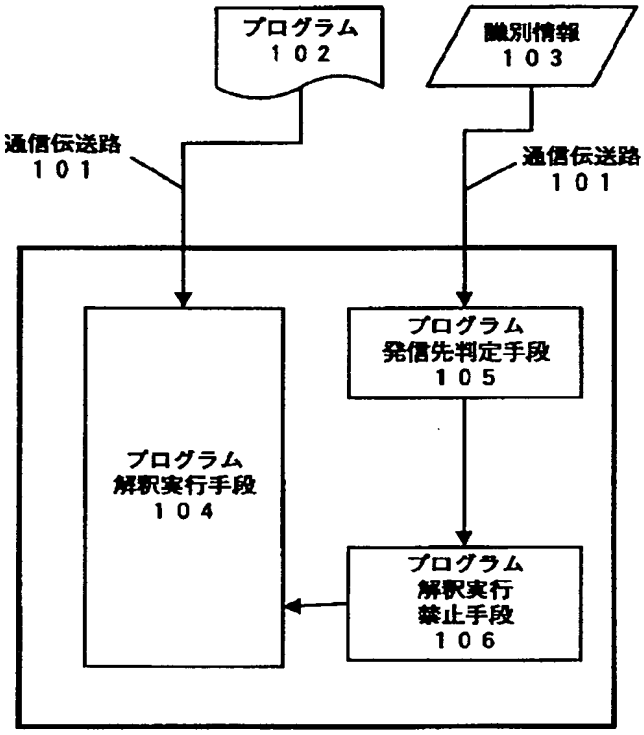
【図３】本発明のプログラム実行装置の構成例を示す図

【図４】本発明のプログラム実行装置の構成例を示す図

【図５】本発明のプログラム実行装置の動作例を示すフ

ローチャート
【図６】本発明のプログラム実行装置の動作例を示すフローチャート
【図７】本発明のプログラム実行装置の動作例を示すフローチャート
【図８】本発明のプログラム実行装置の動作例を示すフローチャート
【図９】プログラム解釈実行禁止手段の構成例を示す図
【図１０】プログラム解釈実行許可確認手段による表示例を示す図
【図１１】プログラム解釈実行選択手段の構成例を示す＊図
【図１２】プログラム解釈実行禁止選択手段の構成例を示す図
【符号の説明】
１０１　通信伝送路
１０２　プログラム
１０３　識別情報
１０４　プログラム解釈実行手段
１０５　プログラム発信先判定手段
１０６　プログラム解釈実行禁止手段

【図１】

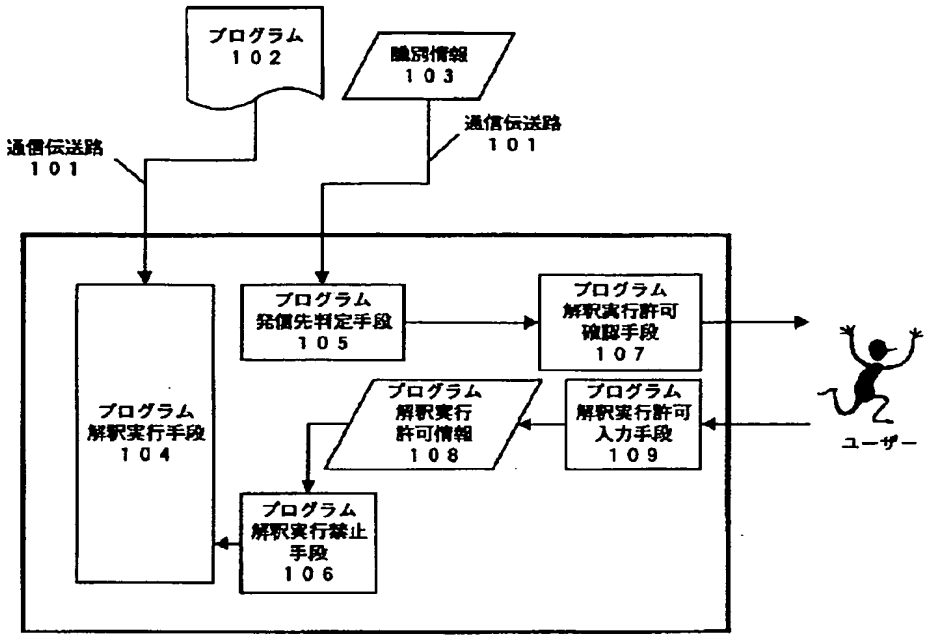



プログラム実行装置・その１

【図１０】


プログラム解釈実行許可確認手段の実施例

【図２】



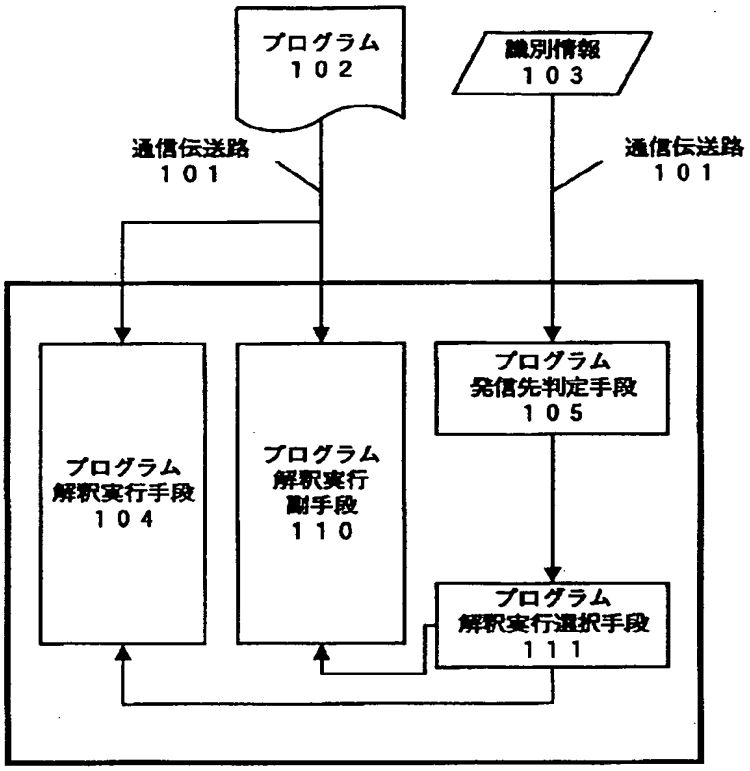

プログラム実行装置・その２

【図９】

| プログラムの識別子 | プログラムの状態 |
|---|---|
| ButtonCrusher.cj | forbidden |
| Maze.cj | permitted |
| ClipBoard.cj | permitted |
| AddressWriter.cj | forbidden |
| DigitalClock.cj | permitted |
| ⋮ | ⋮ |

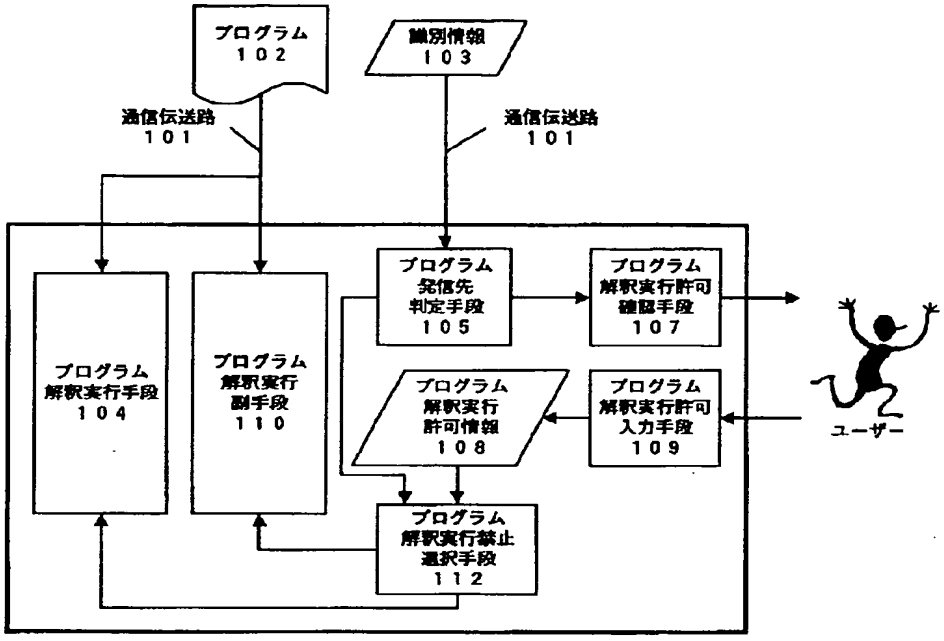
プログラム解釈実行禁止手段の実施例

【図３】




プログラム実行装置・その３

【図４】



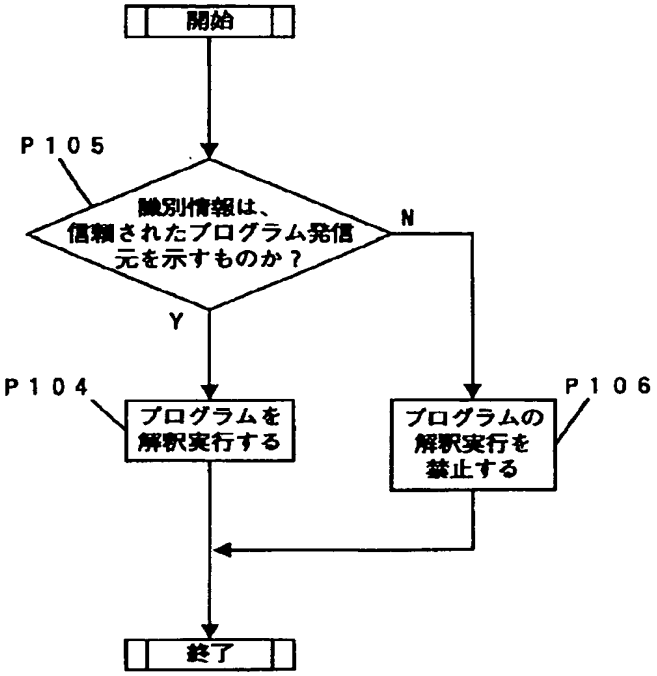

プログラム実行装置・その４

【図１１】

| プログラムの識別子 | プログラムの状態 |
|---|---|
| ButtonCrusher.cj | secured |
| Maze.cj | unsecured |
| ClipBoard.cj | unsecured |
| AddressWriter.cj | secured |
| DigitalClock.cj | unsecured |
| ⋮ | ⋮ |

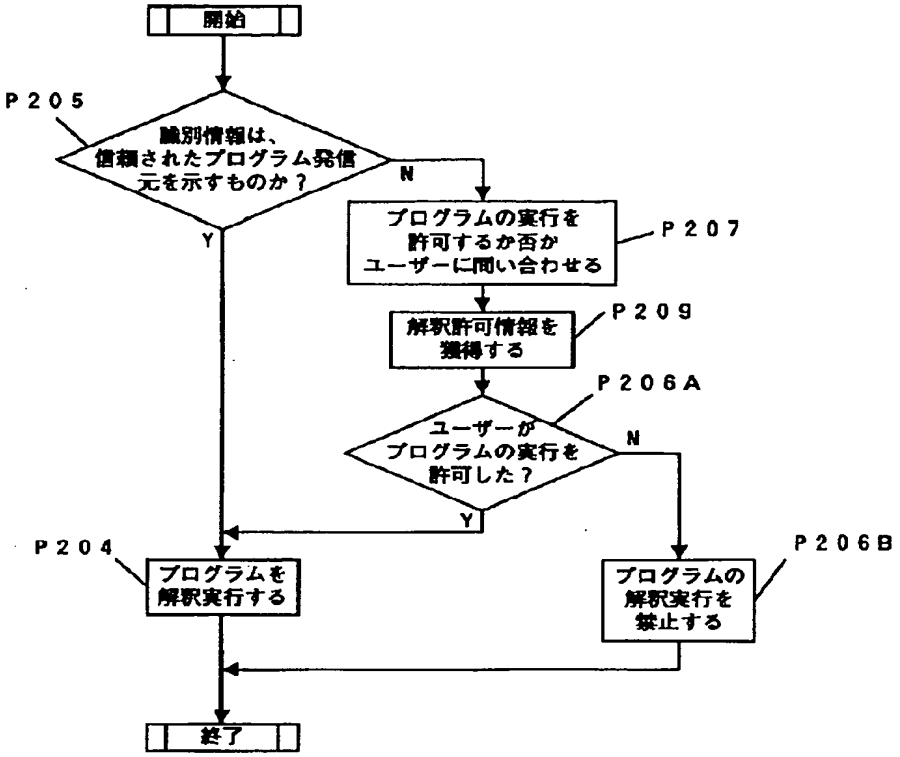
プログラム解釈実行選択手段の実施例

【図５】



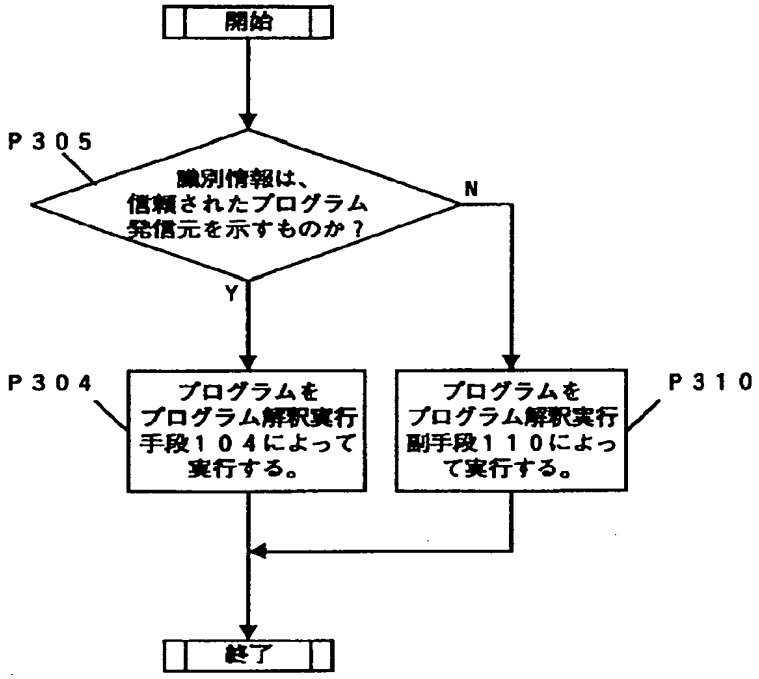

プログラム実行方式の処理手順・その１

【図６】



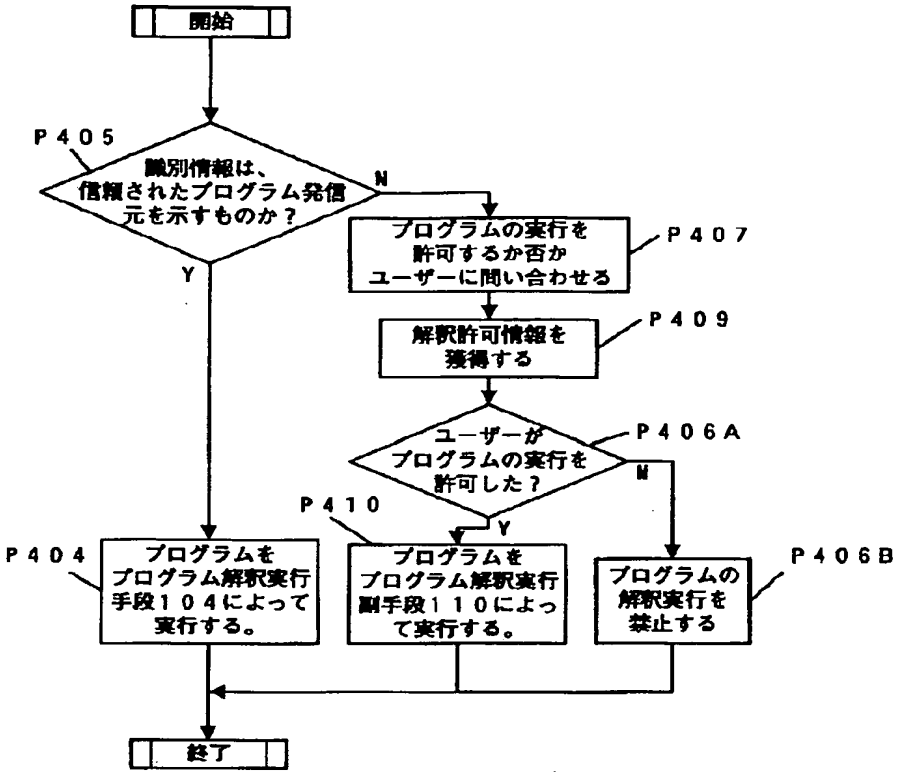

プログラム実行方式の処理手順・その２

【図７】




プログラム実行方式の処理手順・その３

【図８】




プログラム実行方式の処理手順・その４

【図１２】

| プログラムの識別子 | プログラムの状態 |
|---|---|
| ButtonCrusher.cj | forbidden |
| Maze.cj | secured |
| ClipBoard.cj | permitted |
| AddressWriter.cj | forbidden |
| DigitalClock.cj | secured |
| ⋮ | ⋮ |

プログラム解釈実行禁止選択手段の実施例